



Ugrađeni operacijski sustavi

Ugrađeni sustav

- Pojam ugrađeni sustav odnosi se na upotrebu elektronike i softvera unutar proizvoda koji ima određenu funkciju ili skup funkcija, za razliku od računala opće namjene
- Ugrađeni sustav također se može definirati kao bilo koji uređaj koji uključuje računalni čip, ali to nije radna stanica opće namjene ili stolno ili prijenosno računalo
- Primjeri vrsta uređaja s ugrađenim sustavima uključuju mobitele, digitalne kamere, video kamere, kalkulatore, kućne sigurnosne sustave, perilice rublja, razne automobilske sustave, četkice za zube i brojne vrste senzora i aktuatora u automatiziranim sustavima

Ugrađeni sustav

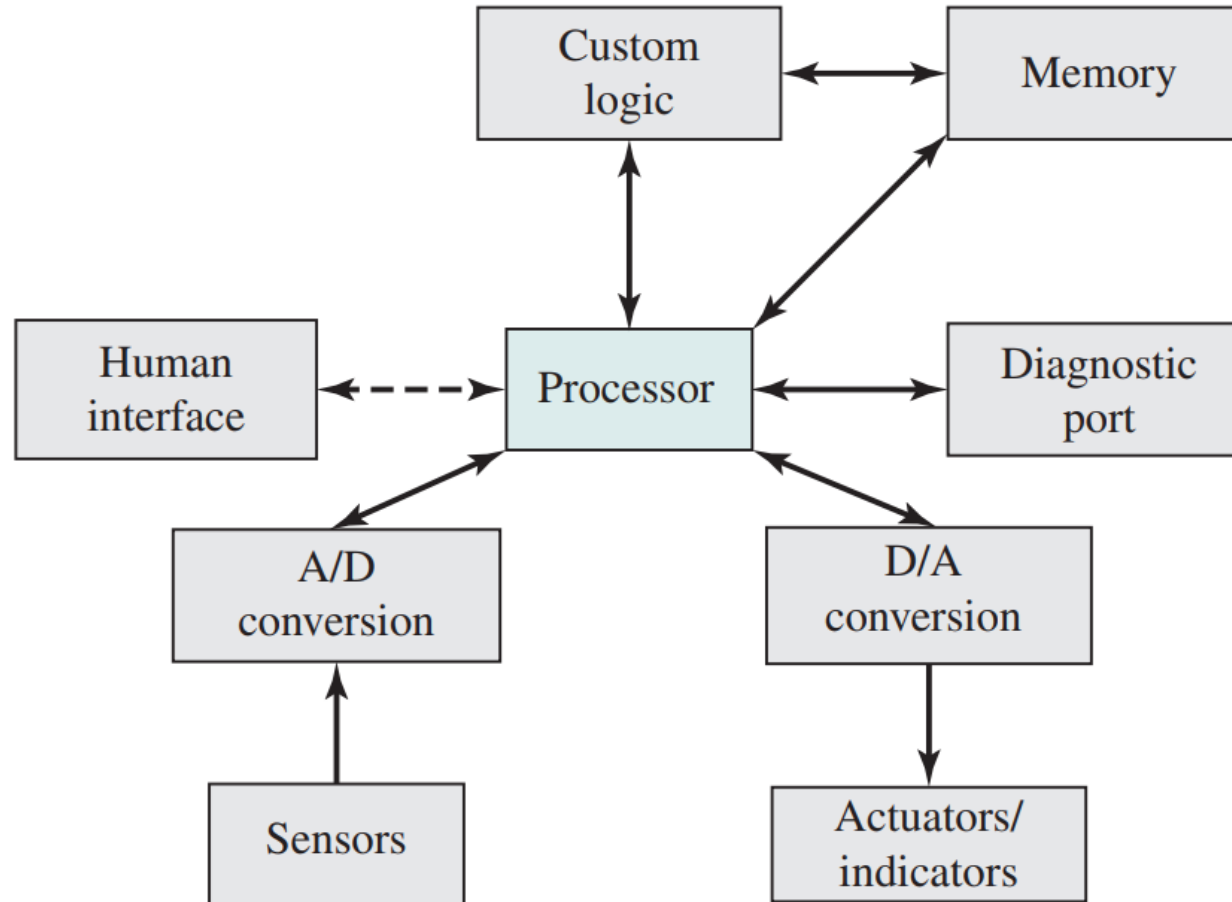
Često su ugrađeni sustavi
čvrsto povezani s
okolinom

Ograničenja, kao što su
potrebna brzina, potrebna
preciznost mjerenja i
potrebno vremensko
trajanje, diktiraju vrijeme
softverskih operacija

To može dovesti do
ograničenja u potrebama
za interakcijom s okolinom
u stvarnom vremenu

Ako se istovremeno mora
upravljati s više aktivnosti,
to nameće složenija
ograničenja u stvarnom
vremenu

Moguća organizacija ugrađenog sustava



Aplikacijski vs namjenski procesori

Aplikacijski procesori

- Definirano mogućnošću procesora da izvršava složene operativne sustave, kao što su Linux, Android i Chrome
- Opća svrha
- Dobar primjer korištenja ugrađenog aplikacijskog procesora je pametni telefon
 - Ugrađeni sustav dizajniran je za podršku brojnim aplikacijama i obavljanje širokog spektra funkcija

Namjenski procesor

- Posvećeno jednom ili malom broju specifičnih zadataka koje zahtijeva glavni uređaj
- Budući da je takav ugrađeni sustav posvećen određenom zadatku ili zadacima, procesor i povezane komponente mogu se prilagoditi u izradi kako bi se smanjila veličina i troškovi

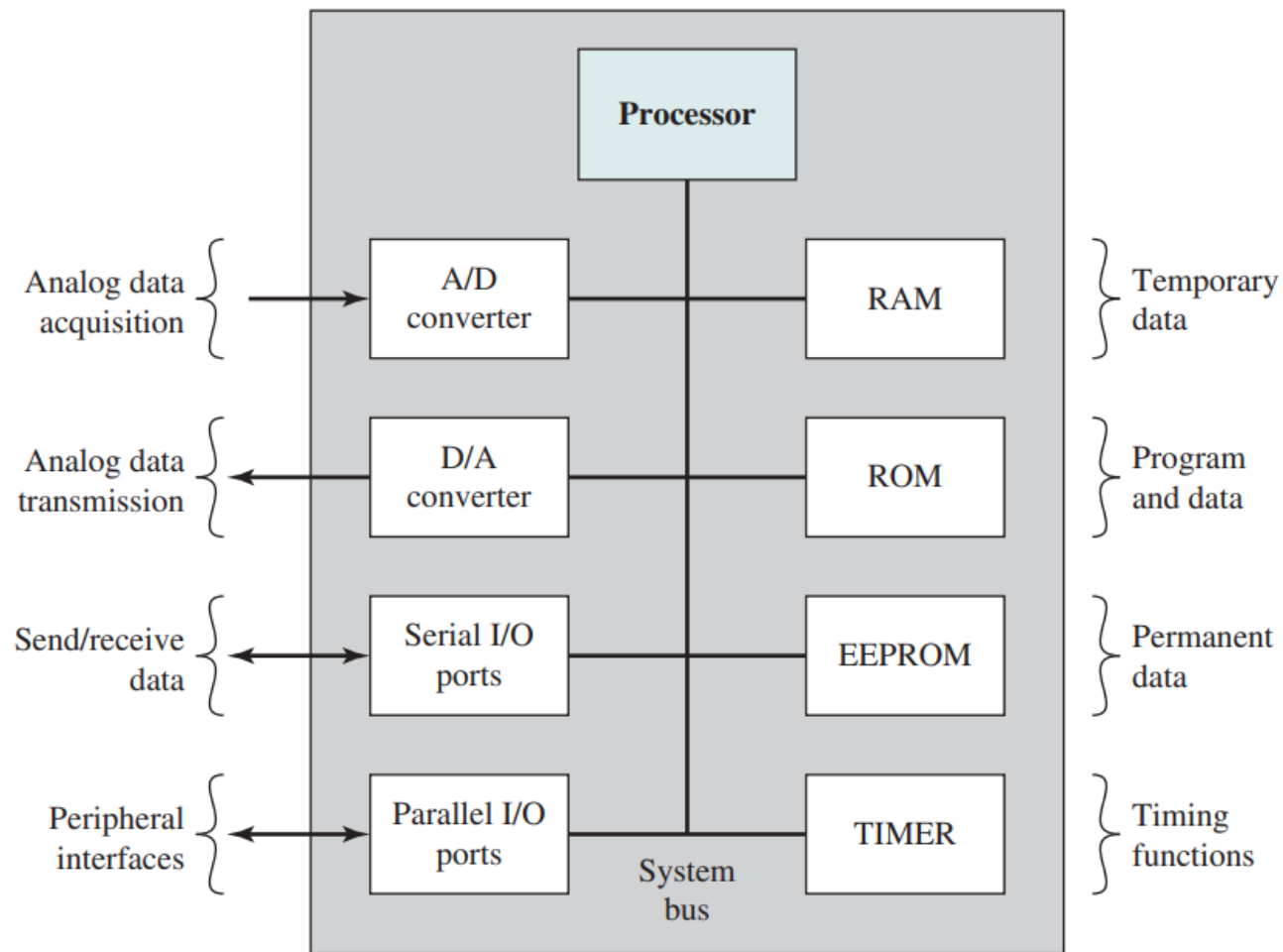
Mikroprocesori

- Mikroprocesor je procesor čiji su elementi minijaturizirani u jedan ili nekoliko integriranih krugova
- Rani mikroprocesorski čipovi uključivali su registre, aritmetičku logičku jedinicu (ALU) i neku vrstu upravljačke jedinice ili logike obrade uputa
- Suvremeni mikroprocesorski čipovi uključuju više procesora, nazvanih jezgrama, i značajnu količinu predmemorske memorije
- Međutim, mikroprocesorski čip uključuje samo neke elemente koji čine računalni sustav

Mikroprocesori

- Tiskana pločica (PCB)
 - Kruta, ravna ploča koja drži i međusobno povezuje čipove i druge elektroničke komponente
 - Sastoji se od slojeva, obično od dva do deset, koji međusobno povezuju komponente putem bakrenih putova koji su urezani u ploču
- Matična ploča
 - Glavni PCB u računalu
 - Naziva se i sistemska ploča
- Ploča za proširenje
 - Manji PCB koji se priključuje na utore na glavnoj ploči
- Čip
 - Najistaknutiji element na matičnoj ploči
 - Jedan komad poluvodičkog materijala, obično silicij, na kojem su izrađeni elektronički krugovi i logička vrata
 - Dobiveni proizvod naziva se integriranim krugom
- Procesor s više jezgri
 - Matična ploča sadrži utor ili utičnicu za procesorski čip, koji obično sadrži više pojedinačnih jezgri, u onome što je poznato kao višejezgreni procesor

Tipični elementi mikrokontrolera



Duboko ugrađen sustav

- Duboko ugrađen sustav ima procesor čije je ponašanje teško promatrati
- Duboko ugrađen sustav:
 - Koristi mikrokontroler, a ne mikroprocesor
 - Ne može se programirati nakon što se programska logika za uređaj spremi u ROM (čitaj samo memoriju)
 - Nema interakciju s korisnikom

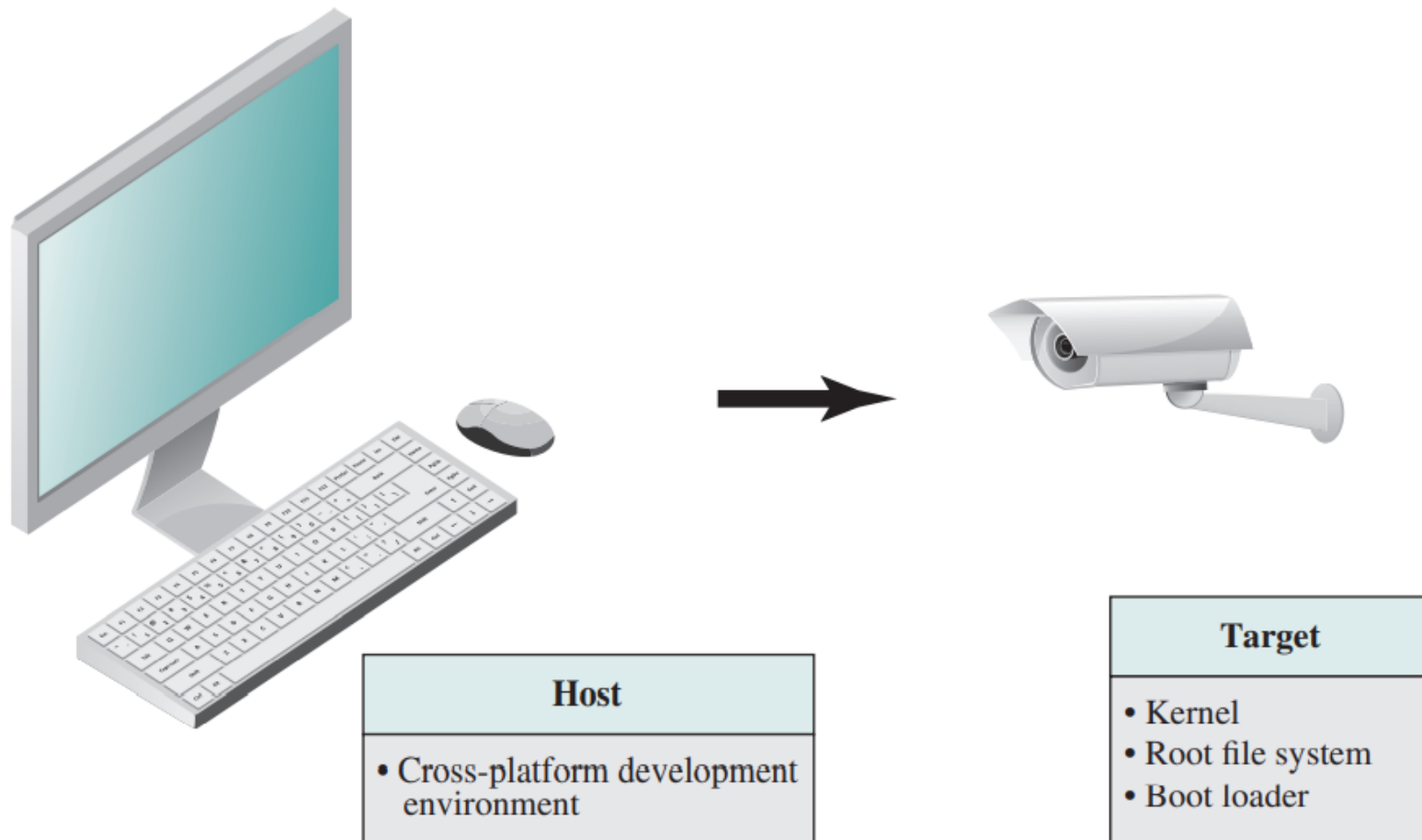
Duboko ugrađen sustav

- Duboko ugrađeni sustavi su jednonamjenski uređaji koji otkrivaju nešto u okolini, obavljaju osnovnu razinu obrade, a zatim rade nešto s rezultatima
- Često imaju bežične mogućnosti i pojavljuju se u umreženim konfiguracijama, kao što su mreže senzora raspoređenih na velikom području
- IoT uvelike ovisi o duboko ugrađenim sustavima
- Obično duboko ugrađeni sustavi imaju ekstremna ograničenja resursa u smislu memorije, veličine procesora, vremena i potrošnje energije

Karakteristike ugrađenog OS-a

- Operacija u stvarnom vremenu
- Reaktivna operacija
- Podesivost
- Fleksibilnost ulazno-izlaznog uređaja
- Pojednostavljeni mehanizmi zaštite
- Izravna uporaba prekida

Host-Target okruženje



Host-Target okruženje

Jezgra

- Puna jezgra uključuje niz zasebnih modula, uključujući:
 - Upravljanje memorijom
 - Upravljanje procesima/dretvama
 - Inter procesna komunikacija, tajmeri
 - Upravljački programi uređaja za ulazno/izlazno računalo, mrežu, zvuk, pohranu, grafiku itd.
 - Datotečni sustavi
 - Umrežavanje
 - Upravljanje energijom

Boat loader

- Mali program koji poziva OS u memoriju (RAM) nakon uključivanja
- Odgovoran je za početni proces pokretanja sustava i za učitavanje jezgre u glavnu memoriju

Root file system

- Sadrži sve datoteke potrebne za rad sustava
- Datotečni sustav ugrađenog OS-a sličan je sustavu koji se nalazi na radnoj stanici ili poslužitelju, osim što sadrži samo minimalni skup aplikacija, biblioteka i povezanih datoteka potrebnih za pokretanje sustava

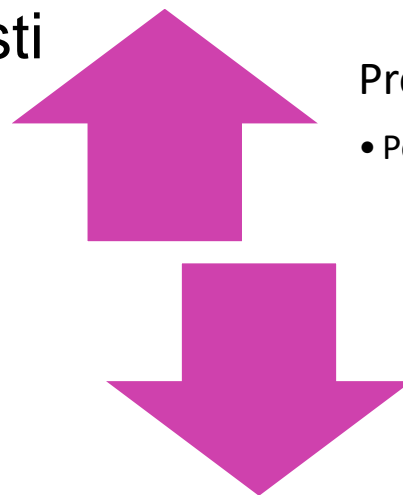
Razvoj ugrađenog OS-a

Dva pristupa:

- Uzmite postojeći OS i prilagodite ga ugrađenoj aplikaciji
- Projektiranje i implementacija OS-a namijenjenog isključivo za ugrađenu uporabu

Prilagodba postojećeg OS-a

- Postojeći komercijalni OS može se koristiti za ugrađeni sustav dodavanjem:
 - Rad u stvarnom vremenu
 - Racionalizacija
 - Dodavanje potrebne funkcionalnosti



Prednosti:

- Poznato sučelje

Nedostataci:

- Nije optimizirano za aplikacije u stvarnom vremenu i ugrađene aplikacije

Namjenski ugrađeni OS

- Karakteristike:

- Brza promjena procesa i dretvi
- Politika planiranja je u stvarnom vremenu, a dispečerski modul je dio planera
- Mala veličina
- Brzo reagira na vanjske prekide
- Minimizira intervale tijekom kojih su prekidi onemogućeni
- Pruža fiksne particije ili particije promjenjive veličine za upravljanje memorijom
- Podržava posebne datoteke koje mogu brzo spremati podatke

Ograničenja vremena

Da biste se nosili s vremenskim ograničenjima, jezgra:

- Pruža ograničeno vrijeme izvršenja
- Održava sat u stvarnom vremenu
- Osigurava posebne alarme i vremenska ograničenja
- Podržava redove čekanja u stvarnom vremenu
- Pruža dretvama odgodu obrade na određeno vrijeme i obustavu/nastavak izvršenja

Ugrađeni Linux

- Verzija Linuxa koja se izvodi u ugrađenom sustavu
- Ugrađeni uređaji obično zahtijevaju podršku za određeni skup uređaja, periferala i protokola, ovisno o hardveru koji je prisutan u određenom uređaju i namjeni tog uređaja
- Ugrađena Linux distribucija verzija je Linuxa koja se prilagođava veličini i hardverskim ograničenjima ugrađenih uređaja
 - Uključuje softverske pakete koji podržavaju razne usluge i aplikacije na tim uređajima
 - Ugrađena Linux jezgra bit će daleko manja od obične Linux jezgre

Datotečni sustavi u ugrađenom Linux-u

- Datotečni sustav mora biti što manji
- Najčešće korišteni primjeri:
 - cramfs
 - Jednostavan datotečni sustav samo za čitanje
 - Datoteke se komprimiraju u jedinicama koje odgovaraju veličini Linux stranice
 - squashfs
 - Komprimirani datotečni sustav samo za čitanje koji je dizajniran za upotrebu u okruženjima s malim brojem memorije ili ograničenom veličinom pohrane
 - jffs2
 - Datotečni sustav temeljen na zapisniku koji je dizajniran za upotrebu na NOR i NAND flash uređajima s posebnom pažnjom na flash-orijentirana pitanja kao što je izravnavanje trošenja
 - ubifs
 - Pruža bolje performanse na većim flash uređajima, a podržava i predmemoriranje pisanja kako bi se osigurale dodatne performance
 - yaffs2
 - Pruža brz i robustan datotečni sustav za velike flash uređaje

Prednosti ugrađenog Linuxa

- Prednosti korištenja Linuxa kao osnove za ugrađeni OS uključuju sljedeće:
 - Neovisnost dobavljača
 - Davatelj platforme ne ovisi o određenom dobavljaču kako bi pružio potrebne značajke i ispunio rokove za implementaciju
 - Raznolika hardverska podrška
 - Linux podrška za širok raspon procesorskih arhitektura i perifernih uređaja čini ga pogodnim za gotovo svaki ugrađeni sustav
 - Niska cijena
 - Korištenje Linuxa smanjuje troškove razvoja i obuke
 - Otvori izvorni kod
 - Korištenje Linuxa pruža sve prednosti softvera otvorenog koda

μ Clinux

- μ Clinux (mikrokontroler Linux) popularna je varijacija Linux jezgre otvorenog koda usmjerena na mikrokontrolere i druge vrlo male ugrađene sustave
- Filozofija dizajna za μ Clinux je smanjiti radno okruženje uklanjanjem uslužnih programa, alata i drugih usluga sustava koje nisu potrebne u ugrađenom okruženju

Razlike između μ Clinuxa i Linuxa

- Linux je multiuser OS temeljen na Unixu. μ Clinux je verzija Linuxa namijenjena ugrađenim sustavima obično bez interaktivnog korisnika
- Za razliku od Linuxa, μ Clinux ne podržava upravljanje memorijom
- Linux jezgra održava zaseban virtualni adresni prostor za svaki proces. μ Clinux ima jedan zajednički adresni prostor za sve procese
- μ Clinux mijenja upravljačke programe uređaja za korištenje lokalne systemske sabirnice, a ne ISA ili PCI
- U Linuxu se adresni prostor oporavlja pri promjeni konteksta; ne kod μ Clinux
- Za razliku od Linuxa, μ Clinux ne pruža poziv funkciji *fork*; jedina opcija je korištenje *vfork*
- Najznačajnija razlika između punog Linuxa i μ Clinuxa je u području upravljanja memorijom

Android

- Fokus Androida leži u vertikalnoj integraciji Linux jezgre i Android korisničkih komponenti
- Mnogi linux programeri ugrađenih sustava ne smatraju Android instancom ugrađenog Linuxa
 - Sa stajališta ovih programera, klasični ugrađeni uređaj ima fiksnu funkciju, definiranu u tvornici

Android

An embedded OS based on a Linux kernel

More of a platform OS that can support a variety of applications that vary from one platform to the next

A vertically integrated system, including some Android specific modifications to the Linux kernel

TinyOS

- Vrlo minimalan OS za ugrađene sustave
- Osnovni OS zahtijeva 400 bajtova koda i podatkovne memorije zajedno
- Ne podržava rad u stvarnom vremenu
- Nema jezgre
- Nema procesa
- OS nema sustav za dodjelu memorije
- Rukovanje prekidima i iznimkama ovisi o perifernim uređajima
- To je potpuno neblokiranje, tako da postoji malo eksplicitnih dretvi za sinkronizaciju
- Popularan pri implementaciji bežičnih senzorskih mreža

Ciljevi TinyOS-a

- Imajući na umu sićušnu distribuiranu aplikaciju senzora, za TinyOS su postavljeni sljedeći ciljevi:
 - Dopusti konkurentnost pri izvršavanju
 - Rad s ograničenim resursima
 - Prilagodljiv razvoju hardvera
 - Podrška širokom broju aplikacija
 - Podrška različitim platformama
 - Robusnost

TinyOS komponente

- Ugrađeni softverski sustavi izgrađeni s TinyOS-om sastoje se od skupa modula, od kojih svaki obavlja jednostavan zadatak i međusobno komuniciraju na ograničene i dobro definirane načine
- Jedini drugi softverski modul je planer
- Budući da nema jezgre, nema stvarnog OS-a
- Područje primjene od interesa je bežična senzorska mreža (WSN)

Primjeri standardiziranih komponenti uključuju:

- Umrežavanje s jednim skokom
- Ad-hoc usmjeravanje
- Upravljanje energijom
- Sat
- Kontrola pohrane

Komponente - Zadaci

- Softverska komponenta implementira jedan ili više zadataka
- Svaki zadatak u komponenti sličan je dretvi u običnom OS-u
- Zadaci su atomarni
 - Nakon pokretanja zadatak se izvršava do kraja

Zadatak ne može:

- Biti prekinut drugim zadatkom
- Biti blokiran ili stavljen na čekanja

Zadatak može:

- Izvoditi izračune
- Pozivanje komponenti niže razine (naredbe)
- Slati signale na višu razinu
- Zakazivati druge zadatke

Komponente - Naredbe

- Naredba je zahtjev koji se ne blokira
 - Zadatak koji izvršava naredbu ne blokira ili ne čeka na odgovor iz komponenti niže razine
- Obično je zahtjev za komponentu niže razine za obavljanje neke usluge
- Učinak na komponentu koja prima naredbu specifičan je za danu naredbu i zadatak potreban za zadovoljavanje naredbe
- Naredba ne može preduhitriti trenutno pokrenut zadatak
- Naredba ne uzrokuje prekidanje u pozvanoj komponenti i ne uzrokuje blokiranje u komponenti pozivanja

Komponente - Događaji

- Događaji u TinyOS-u mogu biti izravno ili neizravno povezani s hardverskim događajima
- Sučelje softverskih komponenti najniže razine izravno na hardverske prekide
 - Mogu biti vanjski prekidi, događaji sata ili događaji brojača
- Rukovatelj događajima u komponenti najniže razine može sam rukovati prekidom ili može prenositi poruke događaja kroz hijerarhiju komponenti
- Naredba može objaviti zadatak koji će signalizirati događaj u budućnosti
 - U ovom slučaju ne postoji veza bilo koje vrste s hardverskim događajem

TinyOS Planer

- Djeluje na svim komponentama
- Istovremeno se izvršava samo jedan zadatak
- Planer je zasebna komponenta
 - To je dio TinyOS-a koji mora biti prisutan u bilo kojem sustavu
- Zadani planer je jednostavan FIFO red čekanja
- Planer je svjestan potrošnje energije
 - Stavlja procesor u stanje mirovanja kada u redu čekanja nema zadatka

Sučelje resursa TinyOS-a

- TinyOS pruža jednostavan, ali moćan skup konvencija za rad s resursima

- **Posvećen**

- Resurs kojem podsustav treba ekskluzivan pristup u svakom trenutku
- Pravila zajedničkog korištenja nisu potrebna
- Primjeri uključuju prekide i brojače

- **Virtualiziran**

- Svaki klijent virtualiziranog resursa komunicira s njim kao da je namjenski resurs
- Primjer je sat ili mjerač vremena

- **Zajednički**

- Apstrakcija koja omogućuje pristup namjenskom resursu putem komponente arbitra
- Arbitar određuje koji klijent ima pristup resursu u koje vrijeme

Sažetak

- Ugrađeni sustavi
 - Koncepti ugrađenog sustava
 - Aplikacijski u odnosu na namjenske procesore
 - Mikroprocesori
 - Mikrokontroleri
 - Duboko ugrađeni sustavi
- Karakteristike ugrađenih operativnih sustava
 - Host-Target okruženje
 - Razvojni pristupi
 - Prilagodba postojećeg komercijalnog operativnog sustava
 - Namjenski ugrađeni operativni sustav
- Ugrađeni Linux
 - Karakteristike ugrađenog Linux sustava
 - Ugrađeni Linux datotečni sustavi
 - Prednosti ugrađenog Linuxa
 - μ Clinux
 - Android
- TinyOS
 - Bežične senzorske mreže
 - Ciljevi TinyOS-a
 - Komponente TinyO-a
 - TinyOS planer
 - Sučelje resursa TinyOS-a



**Thank you for
your attention!**